Sisteme de Gestiune a Bazelor de Date

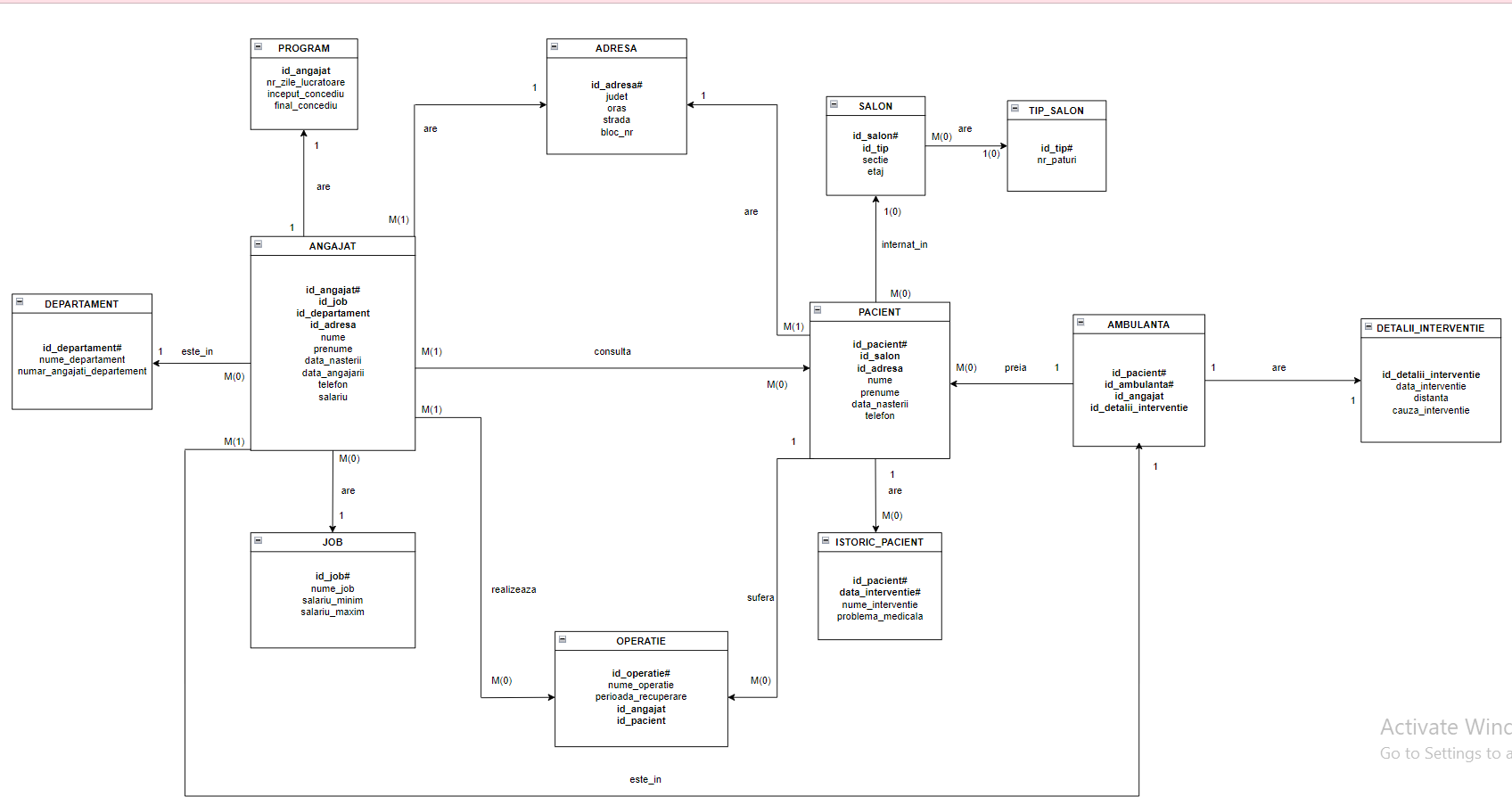
Informatică an II

Tufă Liliana-Ionela, Grupa 234

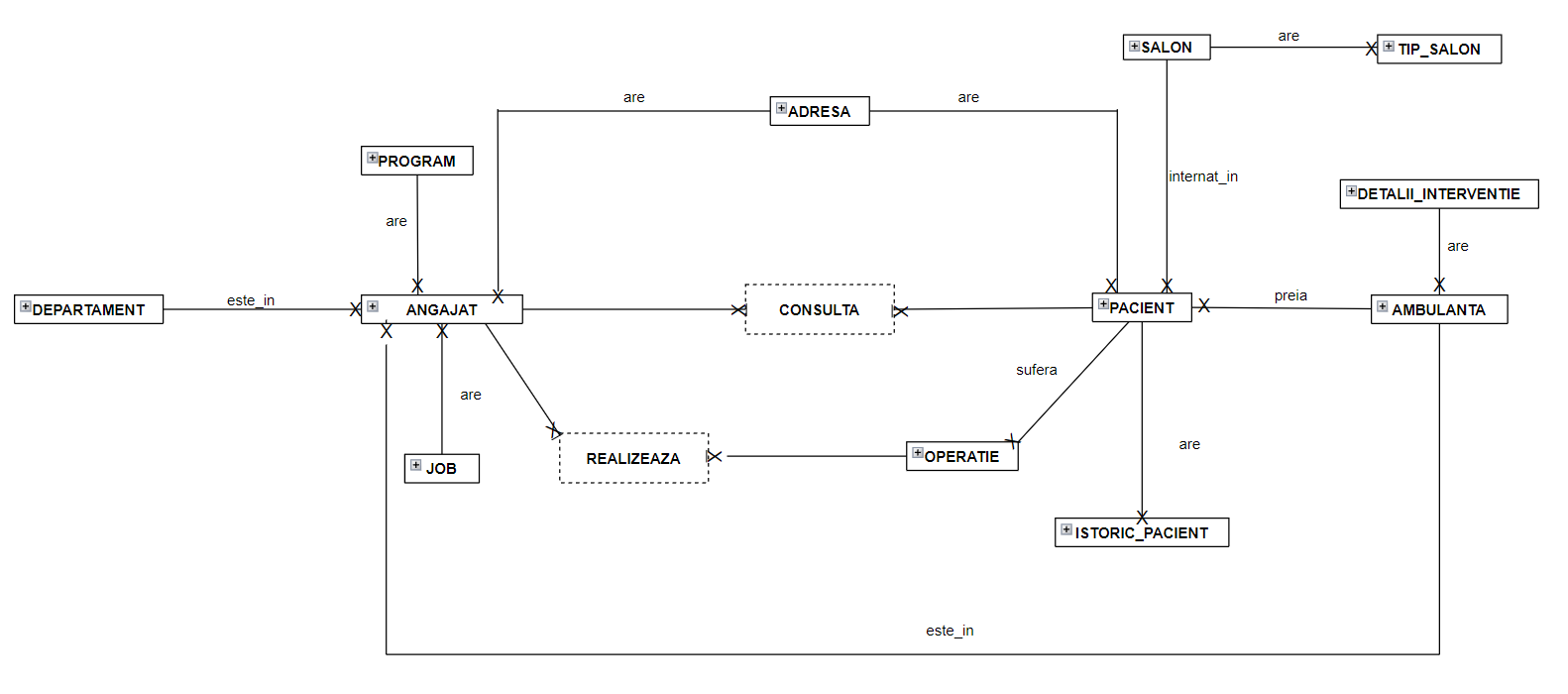
PROIECT

1. **Prezentați pe scurt baza de date (utilitatea ei).**

Baza de date va gestiona informaţii legate de funcționarea unui spital. Sunt stocate informații despre angajați, pacienți, dar și procedurile prin care aceștia trec, cum ar fi consultații sau operații.

1. **Realizați diagrama entitate-relație (ERD).**

1. **Pornind de la diagrama entitate-relație realizați diagrama conceptuală a modelului propus, integrând toate atributele necesare.**

****

1. **Implementați în Oracle diagrama conceptuală realizată: definiți toate tabelele, implementând toate constrângerile de integritate necesare (chei primare, cheile externe etc).**

CREATE TABLE JOB(

id\_job NUMBER(5) PRIMARY KEY NOT NULL,

nume\_job VARCHAR(25) NOT NULL,

salariu\_minim NUMBER(6) NOT NULL,

salariu\_maxim NUMBER(6) NOT NULL

);

CREATE TABLE TIP\_SALON (

id\_tip NUMBER(1) PRIMARY KEY NOT NULL,

nr\_paturi NUMBER(2) NOT NULL

);

CREATE TABLE ADRESA (

id\_adresa NUMBER(5) PRIMARY KEY NOT NULL,

judet VARCHAR(25) NOT NULL,

oras VARCHAR(25) NOT NULL,

strada VARCHAR(25) NOT NULL,

bloc\_nr NUMBER(3) NOT NULL

);

CREATE TABLE DEPARTAMENT (

id\_departament NUMBER(5) PRIMARY KEY NOT NULL,

nume\_departament VARCHAR(50) NOT NULL,

numar\_angajati\_departament NUMBER(3)

);

CREATE TABLE ANGAJAT (

id\_angajat NUMBER(5) PRIMARY KEY NOT NULL,

id\_job NUMBER(5) REFERENCES JOB(id\_job) ON DELETE SET NULL,

id\_departament NUMBER(5) REFERENCES DEPARTAMENT(id\_departament) ON DELETE SET NULL,

id\_adresa NUMBER(5) REFERENCES ADRESA(id\_adresa) ON DELETE SET NULL,

nume VARCHAR(25) NOT NULL,

prenume VARCHAR(25) NOT NULL,

data\_angajarii DATE NOT NULL,

telefon VARCHAR(10) NOT NULL,

salariu NUMBER(6) NOT NULL

);

CREATE TABLE SALON(

id\_salon NUMBER(3) PRIMARY KEY NOT NULL,

id\_tip NUMBER(1) REFERENCES TIP\_SALON(id\_tip) ON DELETE SET NULL,

sectie VARCHAR(50) NOT NULL,

etaj NUMBER(2) NOT NULL

);

CREATE TABLE PROGRAM(

id\_angajat NUMBER(5) REFERENCES ANGAJAT(id\_angajat) ON DELETE SET NULL,

nr\_zile\_lucratoare NUMBER(3) NOT NULL,

inceput\_concediu DATE NOT NULL,

final\_concediu DATE NOT NULL

);

CREATE TABLE PACIENT (

id\_pacient NUMBER(5)PRIMARY KEY NOT NULL,

id\_salon NUMBER(3) REFERENCES SALON(id\_salon),

id\_adresa NUMBER(5) REFERENCES ADRESA(id\_adresa) ON DELETE SET NULL,

nume VARCHAR(25) NOT NULL,

prenume VARCHAR(25) NOT NULL,

data\_nasterii DATE NOT NULL,

telefon VARCHAR(10) NOT NULL

);

CREATE TABLE DETALII\_INTERVENTIE(

id\_detalii\_interventie NUMBER(5) PRIMARY KEY NOT NULL,

data\_interventie DATE NOT NULL,

distanta NUMBER(2) NOT NULL,

cauza\_interventie VARCHAR(50) NOT NULL

);

CREATE TABLE OPERATIE(

id\_operatie NUMBER(5) PRIMARY KEY NOT NULL,

nume\_operatie VARCHAR(50) NOT NULL,

perioada\_recuperare NUMBER(3) NOT NULL,

id\_pacient NUMBER(5) REFERENCES PACIENT(id\_pacient) ON DELETE SET NULL

);

CREATE TABLE ISTORIC\_PACIENT(

id\_pacient NUMBER(5) REFERENCES PACIENT(id\_pacient) NOT NULL,

data\_interventie DATE NOT NULL,

CONSTRAINT PK\_ISTORIC\_PACIENT PRIMARY KEY(id\_pacient, data\_interventie),

nume\_interventie VARCHAR(50) NOT NULL,

problema\_medicala VARCHAR(50) NOT NULL

);

CREATE TABLE AMBULANTA(

id\_pacient NUMBER(5) REFERENCES PACIENT(id\_pacient) NOT NULL,

id\_ambulanta VARCHAR(10) NOT NULL,

CONSTRAINT PK\_AMBULANTA PRIMARY KEY(id\_pacient, id\_ambulanta),

id\_angajat NUMBER(5) REFERENCES ANGAJAT(id\_angajat) NOT NULL,

id\_detalii\_interventie NUMBER(5) REFERENCES DETALII\_INTERVENTIE(id\_detalii\_interventie)

);

CREATE TABLE CONSULTA(

id\_pacient NUMBER(5) REFERENCES PACIENT(id\_pacient) NOT NULL,

id\_angajat NUMBER(5) REFERENCES ANGAJAT(id\_angajat) NOT NULL,

data\_consultatie DATE NOT NULL,

diagnostic VARCHAR(50),

CONSTRAINT PK\_CONSULTA PRIMARY KEY (id\_pacient, id\_angajat)

);

CREATE TABLE REALIZEAZA(

id\_angajat NUMBER(5) REFERENCES ANGAJAT(id\_angajat) NOT NULL,

id\_operatie NUMBER(5) REFERENCES OPERATIE(id\_operatie) NOT NULL,

data\_operatie DATE NOT NULL,

CONSTRAINT PK\_REALIZEZA PRIMARY KEY (id\_operatie, id\_angajat)

);



1. **Adăugați informații coerente în tabelele create (minim 5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru tabela asociativă)**

INSERT INTO ADRESA (id\_adresa, judet, oras, strada, bloc\_nr)

VALUES(14711, 'Teleorman', 'Rosiorii de Vede', 'Aleaa CFR', '5');

INSERT INTO ADRESA (id\_adresa, judet, oras, strada, bloc\_nr)

VALUES(12323, 'Ilfov', 'Bucuresti', 'Serban Voda', '102');

INSERT INTO ADRESA (id\_adresa, judet, oras, strada, bloc\_nr)

VALUES (1041, 'Ilfov', 'Chiajna', 'Tineretului', 31);

INSERT INTO ADRESA (id\_adresa, judet, oras, strada, bloc\_nr)

VALUES (2121, 'Teleorman', 'Videle', 'Republicii', 23);

INSERT INTO ADRESA (id\_adresa, judet, oras, strada, bloc\_nr)

VALUES (14119, 'Suceava', 'Suceava', 'Libertatii', 14);

INSERT INTO ADRESA (id\_adresa, judet, oras, strada, bloc\_nr)

VALUES(564, 'Giurgiu', 'Magura', 'Tineretului', 12);

INSERT INTO ADRESA (id\_adresa, judet, oras, strada, bloc\_nr)

VALUES(23, 'Dambovita', 'Merisor', 'Salcamului', 250);

INSERT INTO ADRESA (id\_adresa, judet, oras, strada, bloc\_nr)

VALUES(67, 'Ilfov', 'Bucuresti', 'Ateneului', 20);

INSERT INTO ADRESA (id\_adresa, judet, oras, strada, bloc\_nr)

VALUES(78, 'Ilfov', 'Bucuresti', 'Cupolei', 23);



INSERT INTO TIP\_SALON (id\_tip, nr\_paturi)

VALUES (1, 1);

INSERT INTO TIP\_SALON (id\_tip, nr\_paturi)

VALUES(2, 2);

INSERT INTO TIP\_SALON (id\_tip, nr\_paturi)

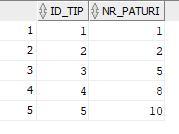
VALUES(3, 5);

INSERT INTO TIP\_SALON (id\_tip, nr\_paturi)

VALUES(4, 8);

INSERT INTO TIP\_SALON (id\_tip, nr\_paturi)

VALUES(5, 10);



INSERT INTO JOB (id\_job, nume\_job, salariu\_minim, salariu\_maxim)

VALUES(5, 'Doctor', 10000, 150000);

INSERT INTO JOB (id\_job, nume\_job, salariu\_minim, salariu\_maxim)

Values(3, 'Asistent', 2500, 5500);

INSERT INTO JOB (id\_job, nume\_job, salariu\_minim, salariu\_maxim)

VALUES(1, 'Personal curatenie', 1400, 2200);

INSERT INTO JOB (id\_job, nume\_job, salariu\_minim, salariu\_maxim)

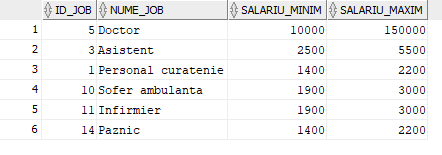
VALUES(10, 'Sofer ambulanta', 1900, 3000);

INSERT INTO JOB (id\_job, nume\_job, salariu\_minim, salariu\_maxim)

VALUES(11, 'Infirmier', 1900, 3000);

INSERT INTO JOB (id\_job, nume\_job, salariu\_minim, salariu\_maxim)

VALUES(14, 'Paznic', 1400, 2200);



INSERT INTO DEPARTAMENT (id\_departament, nume\_departament, numar\_angajati\_departament)

VALUES(1, 'Pediatrie', 3);

INSERT INTO DEPARTAMENT (id\_departament, nume\_departament, numar\_angajati\_departament)

VALUES(2, 'Cardiologie', 2);

INSERT INTO DEPARTAMENT (id\_departament, nume\_departament, numar\_angajati\_departament)

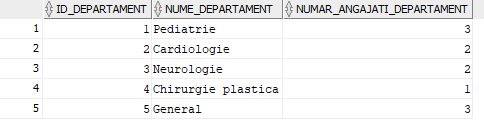
VALUES(3, 'Neurologie', 2);

INSERT INTO DEPARTAMENT (id\_departament, nume\_departament, numar\_angajati\_departament)

VALUES(4, 'Chirurgie plastica', 1);

INSERT INTO DEPARTAMENT (id\_departament, nume\_departament, numar\_angajati\_departament)

VALUES(5, 'General', 3);



INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(3, 5, 1, 14711, 'Olteanu', 'Ana', TO\_DATE ('2015/10/16', 'YYYY/MM//DD'), '0756482944', 15000);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(5, 5, 1, 14711, 'Olteanu', 'Alexandru', TO\_DATE('2015/03/21', 'YYYY/MM/DD'), '076351623', 15000);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(7, 5, 1, 12323, 'Toader', 'Catalin', TO\_DATE('2000/11/23', 'YYYY/MM/DD'), '0762536284', 45000);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(2, 5, 2, 2121, 'Popovici', 'Ioan', TO\_DATE('2003/03/11', 'YYYY/MM/DD'), '0763456782', 40000);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(4, 3, 2, 1041, 'Delcescu', 'Mihai', TO\_DATE('2010/05/11', 'YYYY/MM/DD'), '0789035627', 4000);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(10, 5, 3, 14119, 'Untesu', 'Stefan', TO\_DATE('2005/01/14', 'YYYY/MM/DD'), '0762943123', 150000);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(20, 10, NULL, 1041, 'Popescu', 'Ion', TO\_DATE('2014/11/25', 'YYYY/MM/DD'), '0783647282', 2500);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(21, 10, NULL, 1041, 'Popescu', 'Mircea', TO\_DATE('2017/11/25', 'YYYY/MM/DD'), '0783657282', 2200);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(30, 14, NULL, 78, 'Boicea', 'Serban', TO\_DATE('2019/11/9', 'YYYY/MM/DD'), '0784289423', 2000);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(31, 14, NULL, 67, 'Dolveac', 'Alin', TO\_DATE('2000/11/23', 'YYYY/MM/DD'), '078923342', 2200);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(41, 1, NULL, 23, 'Celeapca', 'Angi', TO\_DATE('2009/10/18', 'YYYY/MM/DD'), '0723456789', 2200);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

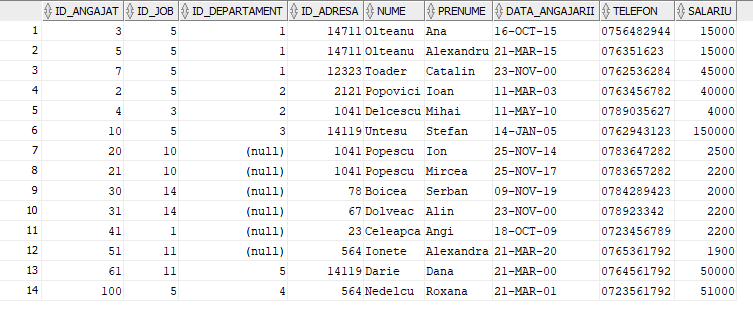
VALUES(51, 11, NULL, 564, 'Ionete', 'Alexandra', TO\_DATE('2020/03/21', 'YYYY/MM/DD'), '0765361792', 1900);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(61, 11, 5, 14119, 'Darie', 'Dana', TO\_DATE('2000/03/21', 'YYYY/MM/DD'), '0764561792', 50000);

INSERT INTO ANGAJAT (id\_angajat, id\_job, id\_departament, id\_adresa, nume, prenume, data\_angajarii, telefon, salariu)

VALUES(100, 5, 4, 564, 'Nedelcu', 'Roxana', TO\_DATE('2001/03/21', 'YYYY/MM/DD'), '0723561792', 51000);



INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(3, 262, TO\_DATE('2021/07/20', 'YYYY/MM/DD'), TO\_DATE('2021/08/04', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(5, 262, TO\_DATE('2021/07/20', 'YYYY/MM/DD'), TO\_DATE('2021/08/04', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(7, 262, TO\_DATE('2021/08/20', 'YYYY/MM/DD'), TO\_DATE('2021/09/04', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(4, 270, TO\_DATE('2021/07/10', 'YYYY/MM/DD'), TO\_DATE('2021/07/24', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(2, 260, TO\_DATE('2021/08/30', 'YYYY/MM/DD'), TO\_DATE('2021/09/14', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(10, 250, TO\_DATE('2021/07/20', 'YYYY/MM/DD'), TO\_DATE('2021/08/14', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(20, 270, TO\_DATE('2021/09/20', 'YYYY/MM/DD'), TO\_DATE('2021/10/04', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(21, 275, TO\_DATE('2021/07/20', 'YYYY/MM/DD'), TO\_DATE('2021/08/04', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

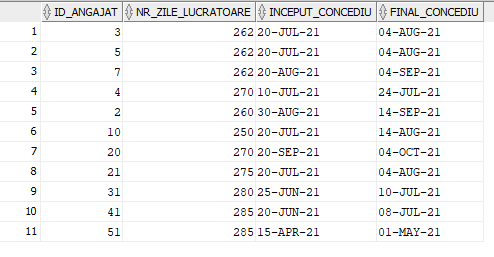
VALUES(31, 280, TO\_DATE('2021/06/25', 'YYYY/MM/DD'), TO\_DATE('2021/07/10', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(41, 285, TO\_DATE('2021/06/20', 'YYYY/MM/DD'), TO\_DATE('2021/07/08', 'YYYY/MM/DD'));

INSERT INTO PROGRAM (id\_angajat, nr\_zile\_lucratoare, inceput\_concediu, final\_concediu)

VALUES(51, 285, TO\_DATE('2021/04/15', 'YYYY/MM/DD'), TO\_DATE('2021/05/01', 'YYYY/MM/DD'));



INSERT INTO SALON (id\_salon, id\_tip, sectie, etaj)

VALUES(10, 1, 'A', 1);

INSERT INTO SALON (id\_salon, id\_tip, sectie, etaj)

VALUES(34, 3, 'B', 3);

INSERT INTO SALON (id\_salon, id\_tip, sectie, etaj)

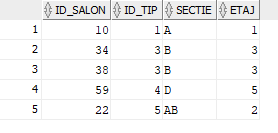
VALUES(38, 3, 'B', 3);

INSERT INTO SALON (id\_salon, id\_tip, sectie, etaj)

VALUES(59, 4, 'D', 5);

INSERT INTO SALON (id\_salon, id\_tip, sectie, etaj)

VALUES(22, 5, 'AB', 2);



INSERT INTO PACIENT (id\_pacient, id\_salon, id\_adresa, nume, prenume, data\_nasterii, telefon)

VALUES(10, 10, 78, 'Alecu', 'Andrei', TO\_DATE('2000/01/01', 'YYYY/MM/DD'), '0765738271');

INSERT INTO PACIENT (id\_pacient, id\_salon, id\_adresa, nume, prenume, data\_nasterii, telefon)

VALUES(11, NULL, 14711, 'Manica', 'Andreea', TO\_DATE('1997/03/01', 'YYYY/MM/DD'), '0765738341');

INSERT INTO PACIENT (id\_pacient, id\_salon, id\_adresa, nume, prenume, data\_nasterii, telefon)

VALUES(12, 34, 564, 'Popescu', 'Andrei', TO\_DATE('1985/01/01', 'YYYY/MM/DD'), '0734738271');

INSERT INTO PACIENT (id\_pacient, id\_salon, id\_adresa, nume, prenume, data\_nasterii, telefon)

VALUES(13, 22, 67, 'Ivanescu', 'David', TO\_DATE('1979/12/25', 'YYYY/MM/DD'), '0769868271');

INSERT INTO PACIENT (id\_pacient, id\_salon, id\_adresa, nume, prenume, data\_nasterii, telefon)

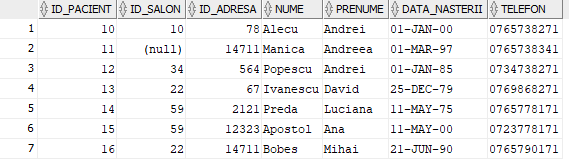
VALUES(14, 59, 2121, 'Preda', 'Luciana', TO\_DATE('1975/05/11', 'YYYY/MM/DD'), '0765778171');

INSERT INTO PACIENT (id\_pacient, id\_salon, id\_adresa, nume, prenume, data\_nasterii, telefon)

VALUES(15, 59, 12323, 'Apostol', 'Ana', TO\_DATE('2000/05/11', 'YYYY/MM/DD'), '0723778171');

INSERT INTO PACIENT (id\_pacient, id\_salon, id\_adresa, nume, prenume, data\_nasterii, telefon)

VALUES(16, 22, 14711, 'Bobes', 'Mihai', TO\_DATE('1990/06/21', 'YYYY/MM/DD'), '0765790171');



INSERT INTO DETALII\_INTERVENTIE (id\_detalii\_interventie, data\_interventie, distanta, cauza\_interventie)

VALUES(100, TO\_DATE('2021/06/01', 'YYYY/MM/DD'), 45, 'atac cerebral');

INSERT INTO DETALII\_INTERVENTIE (id\_detalii\_interventie, data\_interventie, distanta, cauza\_interventie)

VALUES(101, TO\_DATE('2021/06/02', 'YYYY/MM/DD'), 5, 'arsuri');

INSERT INTO DETALII\_INTERVENTIE (id\_detalii\_interventie, data\_interventie, distanta, cauza\_interventie)

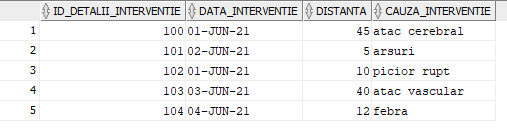
VALUES(102, TO\_DATE('2021/06/01', 'YYYY/MM/DD'), 10, 'picior rupt');

INSERT INTO DETALII\_INTERVENTIE (id\_detalii\_interventie, data\_interventie, distanta, cauza\_interventie)

VALUES(103, TO\_DATE('2021/06/03', 'YYYY/MM/DD'), 40, 'atac vascular');

INSERT INTO DETALII\_INTERVENTIE (id\_detalii\_interventie, data\_interventie, distanta, cauza\_interventie)

VALUES(104, TO\_DATE('2021/06/04', 'YYYY/MM/DD'), 12, 'febra');



INSERT INTO AMBULANTA (id\_pacient, id\_ambulanta, id\_angajat, id\_detalii\_interventie)

VALUES(16, 'B23AMB', 20, 100);

INSERT INTO AMBULANTA (id\_pacient, id\_ambulanta, id\_angajat, id\_detalii\_interventie)

VALUES(11, 'B24AMB', 51, 101);

INSERT INTO AMBULANTA (id\_pacient, id\_ambulanta, id\_angajat, id\_detalii\_interventie)

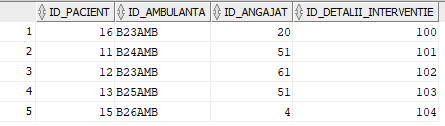
VALUES(12, 'B23AMB', 61, 102);

INSERT INTO AMBULANTA (id\_pacient, id\_ambulanta, id\_angajat, id\_detalii\_interventie)

VALUES(13, 'B25AMB', 51, 103);

INSERT INTO AMBULANTA (id\_pacient, id\_ambulanta, id\_angajat, id\_detalii\_interventie)

VALUES(15, 'B26AMB', 4, 104);



INSERT INTO OPERATIE(id\_operatie, nume\_operatie, perioada\_recuperare, id\_pacient)

VALUES(1, 'reconstructie nazala', 15, 11);

INSERT INTO OPERATIE(id\_operatie, nume\_operatie, perioada\_recuperare, id\_pacient)

VALUES(2, 'reconstructie nazala', 15, 15);

INSERT INTO OPERATIE(id\_operatie, nume\_operatie, perioada\_recuperare, id\_pacient)

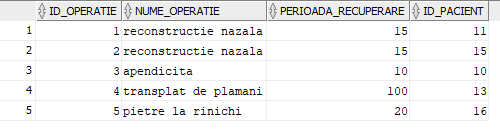
VALUES(3, 'apendicita', 10, 10);

INSERT INTO OPERATIE(id\_operatie, nume\_operatie, perioada\_recuperare, id\_pacient)

VALUES(4, 'transplat de plamani', 100, 13);

INSERT INTO OPERATIE(id\_operatie, nume\_operatie, perioada\_recuperare, id\_pacient)

VALUES(5, 'pietre la rinichi', 20, 16);



INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

VALUES(100, 1, to\_date('2019/11/21 08:30:00', 'YYYY/MM/DD HH:MI:SS'));

INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

VALUES(61, 1, TO\_DATE('2019/11/21', 'YYYY/MM/DD'));

INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

VALUES(51, 1, TO\_DATE('2019/11/21', 'YYYY/MM/DD'));

INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

VALUES(100, 2, TO\_DATE('2019/12/23', 'YYYY/MM/DD'));

INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

VALUES(51, 2, TO\_DATE('2019/12/24', 'YYYY/MM/DD'));

INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

VALUES(61, 3, TO\_DATE('2020/11/25', 'YYYY/MM/DD'));

INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

VALUES(4, 3, TO\_DATE('2020/01/13', 'YYYY/MM/DD'));

INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

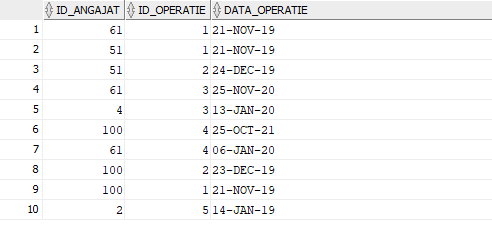
VALUES(100, 4, TO\_DATE('2021/10/25', 'YYYY/MM/DD'));

INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

VALUES(61, 4, TO\_DATE('2020/01/06', 'YYYY/MM/DD'));

INSERT INTO REALIZEAZA(id\_angajat, id\_operatie, data\_operatie)

VALUES(2, 5, TO\_DATE('2019/01/14', 'YYYY/MM/DD'));



insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

values (2, 10, TO\_DATE('2020/01/06', 'YYYY/MM/DD'), 'bronsita');

insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

values (2, 11, TO\_DATE('2020/01/06', 'YYYY/MM/DD'), 'gripa');

insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

values (2, 12, TO\_DATE('2020/01/10', 'YYYY/MM/DD'), 'probleme cardiace');

insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

values (10, 15, TO\_DATE('2020/01/20', 'YYYY/MM/DD'), 'atac cerebral');

insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

values (100, 15, TO\_DATE('2020/01/20', 'YYYY/MM/DD'), 'atac cerebral');

insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

values (10, 14, TO\_DATE('2020/02/06', 'YYYY/MM/DD'), 'probleme vasculare');

insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

values (100, 16, TO\_DATE('2019/06/06', 'YYYY/MM/DD'), 'pietre la rinichi');

insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

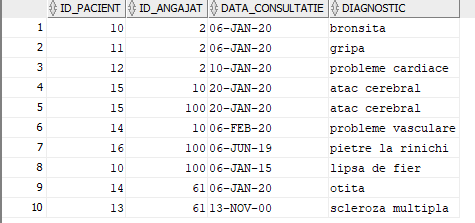
values (100, 10, TO\_DATE('2015/01/06', 'YYYY/MM/DD'), 'lipsa de fier');

insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

values (61, 14, TO\_DATE('2020/01/06', 'YYYY/MM/DD'), 'otita');

insert into CONSULTA (id\_angajat, id\_pacient, data\_consultatie, diagnostic)

values (61, 13, TO\_DATE('2000/11/13', 'YYYY/MM/DD'), 'scleroza multipla');



insert into ISTORIC\_PACIENT (id\_pacient, data\_interventie, nume\_interventie, problema\_medicala)

values (10, TO\_DATE('2019/04/06', 'YYYY/MM/DD'), 'consultatie', 'otita');

insert into ISTORIC\_PACIENT (id\_pacient, data\_interventie, nume\_interventie, problema\_medicala)

values (10, TO\_DATE('2020/11/06', 'YYYY/MM/DD'), 'consultatie', 'bronsita');

insert into ISTORIC\_PACIENT (id\_pacient, data\_interventie, nume\_interventie, problema\_medicala)

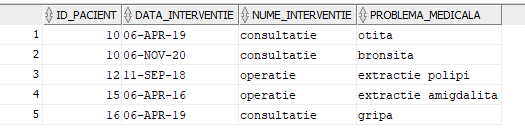
values (12, TO\_DATE('2018/09/11', 'YYYY/MM/DD'), 'operatie', 'extractie polipi');

insert into ISTORIC\_PACIENT (id\_pacient, data\_interventie, nume\_interventie, problema\_medicala)

values (15, TO\_DATE('2016/04/06', 'YYYY/MM/DD'), 'operatie', 'extractie amigdalita');

insert into ISTORIC\_PACIENT (id\_pacient, data\_interventie, nume\_interventie, problema\_medicala)

values (16, TO\_DATE('2019/04/06', 'YYYY/MM/DD'), 'consultatie', 'gripa');



1. **Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat care să utilizeze două tipuri de colecție studiate. Apelați subprogramul.**

--Afisati departamentele cu id impar, ce au mai multi angajati decat un numar nr dat ca parametru.

create or replace procedure ex6(nr number)

is

type tabel\_indexat is table of varchar2(50) index by pls\_integer;

type vector is array(3) of number;

v vector := vector();

t tabel\_indexat;

j number;

begin

j := 0;

for i in 1..5 loop

if i mod 2 = 1 then

j := j + 1;

v.extend;

v(j) := i;

end if;

end loop;

for i in v.first..v.last loop

select nume\_departament into t(i)

from departament

where id\_departament = v(i) and numar\_angajati\_departament >= nr;

end loop;

for i in t.first..t.last loop

dbms\_output.put\_line(t(i));

end loop;

exception

when no\_data\_found then

raise\_application\_error(-20000, 'Nu exista departamente');

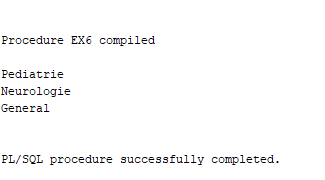
end;

/

begin

ex6(2);

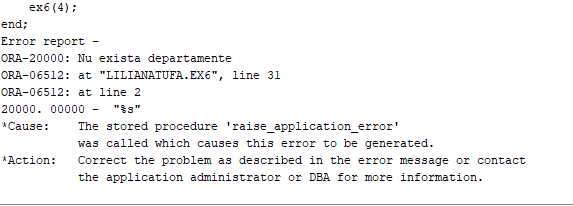
end;



begin

ex6(4);

end;



1. **Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat care să utilizeze un tip de cursor studiat. Apelați subprogramul.**

--Afisati pentru fiecare job numele si prenumele persoanelor angajate dupa un an dat ca parametru.

create or replace procedure ex7(nr number)

is

nr\_val number :=0;

exceptie exception;

begin

for i in (select id\_job, nume\_job from job) loop

for j in (select id\_job, nume, prenume, data\_angajarii from angajat) loop

if i.id\_job = j.id\_job then

if extract(year from j.data\_angajarii) >= nr then

nr\_val := nr\_val + 1;

dbms\_output.put\_line(i.nume\_job || ': ' || j.nume || ' ' || j.prenume);

end if;

end if;

end loop;

dbms\_output.new\_line;

end loop;

if nr\_val = 0 then

raise exceptie;

end if;

exception

when exceptie then

raise\_application\_error(-20000, 'Nu exista personal angajat dupa acest an');

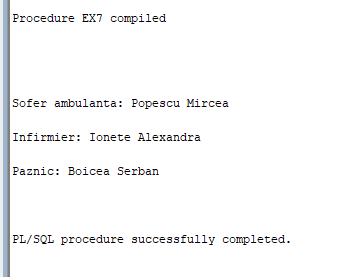
end;

/

begin

ex7(2017);

end;

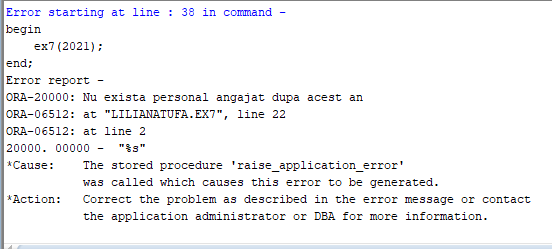


begin

ex7(2021);

end;

/



1. **Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat de tip funcție care să utilizeze într-o singură comandă SQL 3 dintre tabelele definite. Tratați toate excepțiile care pot apărea. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.**

--Determinati numarul de pacienti ce s-au internat in urma unei operatii, numele operatiei fiind dat ca parametru.

create or replace function ex8

(nume\_op operatie.nume\_operatie%type)

return number is

nr number := 0;

exceptie exception;

begin

select count(s.id\_salon)

into nr

from operatie o, pacient p, salon s

where lower(o.nume\_operatie) = lower(nume\_op) and o.id\_pacient = p.id\_pacient and p.id\_salon= s.id\_salon;

if nr = 0 then

raise exceptie;

end if;

return nr;

exception

when exceptie then

raise\_application\_error(-20000, 'Nu s-au internat pacienti');

when others then

raise\_application\_error(-20002, 'Alta eroare');

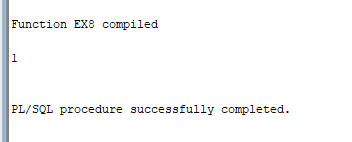
end;

/

begin

dbms\_output.put\_line(ex8('reconstructie nazala'));

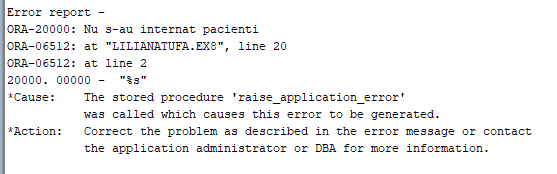
end;



begin

dbms\_output.put\_line(ex8('transplant de plamani'));

end;



1. **Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat de tip procedură care să utilizeze într-o singură comandă SQL 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea, incluzând excepțiile NO\_DATA\_FOUND și TOO\_MANY\_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.**

--Aratati ce probleme medicale au mai avut pacientii (din tabela istoric\_pacient) care au fost operati de un anumit doctor, numele acestuia fiind dat ca parametru

create or replace procedure ex9

(nume\_ang angajat.nume%type)

is

type tabel\_indexat is table of varchar2(50) index by pls\_integer;

informatii tabel\_indexat;

id\_ang angajat.id\_angajat%type;

begin

select id\_angajat into id\_ang

from angajat

where lower(nume) = lower(nume\_ang);

select i.problema\_medicala

bulk collect into informatii

from angajat a, realizeaza r, operatie o, pacient p, istoric\_pacient i

where id\_ang = a.id\_angajat and a.id\_angajat= r.id\_angajat and r.id\_operatie=o.id\_operatie

and o.id\_pacient = p.id\_pacient and p.id\_pacient = i.id\_pacient;

dbms\_output.put\_line('Informatii despre pacientii operati de dr. ' || nume\_ang || ': ');

for i in informatii.first..informatii.last loop

dbms\_output.put\_line(informatii(i));

end loop;

EXCEPTION

when no\_data\_found then

raise\_application\_error(-20000,'Nu exista anagajatul introdus.');

when too\_many\_rows then

raise\_application\_error(-20001,'Exista mai multe informatii ale angajatilor operati de angajatul dat.');

when others then

raise\_application\_error(-20002, 'Alta erorare!!');

end;

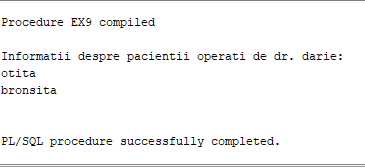
/

begin

ex9('darie');

end;

/

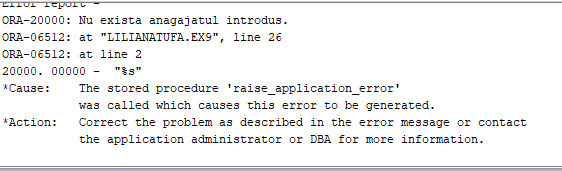


begin

ex9('georgescu');

end;

/

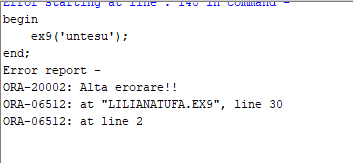


begin

ex9('untesu');

end;

/



1. **Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul.**

--Nu se pot face modificari asupra tabelei job in lunile de iarna ale anului.

create or replace trigger trigger\_1 before

update or insert or delete on job

begin

if to\_char(sysdate, 'MM') = 1 or to\_char(sysdate, 'MM') = 2 or to\_char(sysdate, 'MM') = 12 then

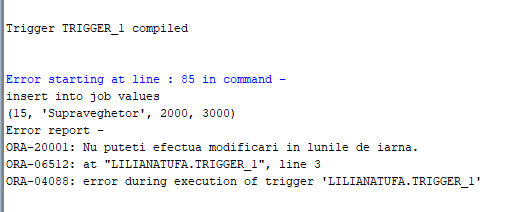
raise\_application\_error(-20001, 'Nu puteti efectua modificari in lunile de iarna.');

end if;

end;

insert into job values

(15, 'Supraveghetor', 2000, 3000);



1. **Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul.**

--Nu puteti modifica data unei consultatii cu o data mai mica decat data curenta.

create or replace trigger trigger\_2 before

update of data\_consultatie on consulta

for each row

begin

if (:new.data\_consultatie < sysdate) then

raise\_application\_error(-20002, 'Nu puteti modifica data consultatiei cu o data din trecut.');

end if;

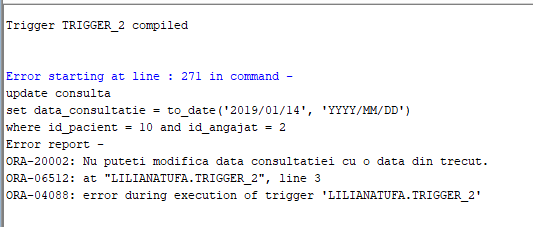
end;

/

update consulta

set data\_consultatie = to\_date('2019/01/14', 'YYYY/MM/DD')

where id\_pacient = 10 and id\_angajat = 2;



1. **Definiți un trigger de tip LDD. Declanșați trigger-ul.**

--Nu puteti sterge tabele din baza de date.

create or replace trigger trigger\_3 before

drop on database

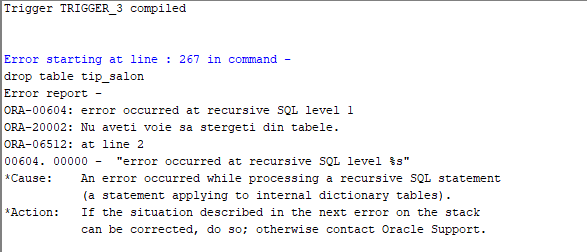
begin

raise\_application\_error(-20002, 'Nu aveti voie sa stergeti din tabele.');

end;

/

drop table tip\_salon;



1. **Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului.**

create or replace package pachet as

procedure ex6 (nr number);

procedure ex7 (nr number);

function ex8 (nume\_op operatie.nume\_operatie%type) return number;

procedure ex9 (nume\_ang angajat.nume%type);

end pachet;

/

create or replace package body pachet as

procedure ex6(nr number)

is

type tabel\_indexat is table of varchar2(50) index by pls\_integer;

type vector is array(3) of number;

v vector := vector();

t tabel\_indexat;

j number;

begin

j := 0;

for i in 1..5 loop

if i mod 2 = 1 then

j := j + 1;

v.extend;

v(j) := i;

end if;

end loop;

for i in v.first..v.last loop

select nume\_departament into t(i)

from departament

where id\_departament = v(i) and numar\_angajati\_departament >= nr;

end loop;

for i in t.first..t.last loop

dbms\_output.put\_line(t(i));

end loop;

exception

when no\_data\_found then

raise\_application\_error(-20000, 'Nu exista departamente');

end ex6;

procedure ex7(nr number)

is

nr\_val number :=0;

exceptie exception;

begin

for i in (select id\_job, nume\_job from job) loop

for j in (select id\_job, nume, prenume, data\_angajarii from angajat) loop

if i.id\_job = j.id\_job then

if extract(year from j.data\_angajarii) >= nr then

nr\_val := nr\_val + 1;

dbms\_output.put\_line(i.nume\_job || ': ' || j.nume || ' ' || j.prenume);

end if;

end if;

end loop;

dbms\_output.new\_line;

end loop;

if nr\_val = 0 then

raise exceptie;

end if;

exception

when exceptie then

raise\_application\_error(-20000, 'Nu exista personal angajat dupa acest an');

end ex7;

function ex8

(nume\_op operatie.nume\_operatie%type)

return number is

nr number := 0;

exceptie exception;

begin

select count(s.id\_salon)

into nr

from operatie o, pacient p, salon s

where lower(o.nume\_operatie) = lower(nume\_op) and o.id\_pacient = p.id\_pacient and p.id\_salon= s.id\_salon;

if nr = 0 then

raise exceptie;

end if;

return nr;

exception

when exceptie then

raise\_application\_error(-20000, 'Nu s-au internat pacienti');

when others then

raise\_application\_error(-20002, 'Alta eroare');

end ex8;

procedure ex9

(nume\_ang angajat.nume%type)

is

type tabel\_indexat is table of varchar2(50) index by pls\_integer;

informatii tabel\_indexat;

id\_ang angajat.id\_angajat%type;

begin

select id\_angajat into id\_ang

from angajat

where lower(nume) = lower(nume\_ang);

select i.problema\_medicala

bulk collect into informatii

from angajat a, realizeaza r, operatie o, pacient p, istoric\_pacient i

where id\_ang = a.id\_angajat and a.id\_angajat= r.id\_angajat and r.id\_operatie=o.id\_operatie

and o.id\_pacient = p.id\_pacient and p.id\_pacient = i.id\_pacient;

dbms\_output.put\_line('Informatii despre pacientii operati de dr. ' || nume\_ang || ': ');

for i in informatii.first..informatii.last loop

dbms\_output.put\_line(informatii(i));

end loop;

EXCEPTION

when no\_data\_found then

raise\_application\_error(-20000,'Nu exista anagajatul introdus.');

when too\_many\_rows then

raise\_application\_error(-20001,'Exista mai multe informatii ale angajatilor operati de angajatul dat.');

when others then

raise\_application\_error(-20002, 'Alta erorare!!');

end ex9;

end pachet;

/

begin

pachet.ex6(2);

dbms\_output.new\_line;

pachet.ex7(2017);

dbms\_output.new\_line;

dbms\_output.put\_line('Numarul pacientilor internati este ' || pachet.ex8('reconstructie nazala'));

dbms\_output.new\_line;

pachet.ex9('darie');

end;

/

